

8/5/1 (Item 1 from file: 351)  
DIALOG(R) File 351:Derwent WPI  
(c) 2002 Thomson Derwent. All rts. reserv.

012630316 \*\*Image available\*\*  
WPI Acc No: 1999-436420/ 199937  
XRPX Acc No: N99-325739

Data communication system of electronic apparatus - writes data chosen as  
object of communication from data sources in data descriptor

Patent Assignee: SONY CORP (SONY )

Inventor: KAWAMURA H

Number of Countries: 029 Number of Patents: 005

Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
JP 11177588	A	19990702	JP 97339576	A	19971210	199937 B
EP 935370	A1	19990811	EP 98310073	A	19981209	199937
CN 1219811	A	19990616	CN 98126024	A	19981210	199942
KR 99062904	A	19990726	KR 9853806	A	19981208	200043
US 6360287	B1	20020319	US 98204297	A	19981203	200224

Priority Applications (No Type Date): JP 97339576 A 19971210

Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
-----------	------	-----	----	----------	--------------

JP 11177588	A		10	H04L-012/40	
-------------	---	--	----	-------------	--

EP 935370	A1	E	20	H04L-012/64	
-----------	----	---	----	-------------	--

Designated States (Regional): AL AT BE CH CY DE DK ES FI FR GB GR IE IT

LI LT LU LV MC MK NL PT RO SE SI

CN 1219811	A			H04L-012/40	
------------	---	--	--	-------------	--

KR 99062904	A			H04L-012/58	
-------------	---	--	--	-------------	--

US 6360287	B1			G06F-013/00	
------------	----	--	--	-------------	--

Abstract (Basic): JP 11177588 A

NOVELTY - The data chosen as object of communication from the data  
sources (1) are written in the data descriptor (42). DETAILED  
DESCRIPTION - An INDEPENDENT CLAIM is also included for data  
communication procedure.

USE - For data communication between electronic apparatus such as  
digital video cassette recorder, television receiver, personal  
computer.

ADVANTAGE - Since the format of data descriptor is expanded to  
transmit data by asynchronous packet, developing a new protocol is not  
necessary. DESCRIPTION OF DRAWING(S) - The figure depicts block diagram  
showing data descriptor in the data communication system. (1) Data  
sources; (42) Data descriptor.

Dwg.1/13

Title Terms: DATA; COMMUNICATE; SYSTEM; ELECTRONIC; APPARATUS; WRITING;

DATA; CHOICE; OBJECT; COMMUNICATE; DATA; SOURCE; DATA; DESCRIBE

Derwent Class: W01

International Patent Class (Main): G06F-013/00; H04L-012/40; H04L-012/58;  
H04L-012/64

International Patent Class (Additional): G06F-013/38; H04L-012/56;

H04L-029/06

File Segment: EPI

8/5/2 (Item 1 from file: 347)  
DIALOG(R) File 347:JAPIO  
(c) 2002 JPO & JAPIO. All rts. reserv.

06236017 \*\*Image available\*\*  
ELECTRONIC EQUIPMENT AND DATA COMMUNICATION METHOD

PUB. NO.: 11-177588 A]  
PUBLISHED: July 02, 1999 (19990702)  
INVENTOR(s): KAWAMURA HARUMI

BEST AVAILABLE COPY

APPLICANT(s): SONY CORP  
APPL. NO.: 09-339576 [JP 97339576]  
FILED: December 10, 1997 (19971210)  
INTL CLASS: H04L-012/40; G06F-013/00; G06F-013/38; H04L-012/56;  
H04L-029/06

ABSTRACT

PROBLEM TO BE SOLVED: To transmit small quantity, of data such as still picture data by an asynchronous packet.

SOLUTION: A source equipment 1 writes data selectable from a data source block 11 into a data descriptor in a memory 13. Then, data read-out from the data descriptor is transmitted to a receiver equipment 2 by the asynchronous packet. The source equipment 2 writes data selectable from the data source block 11 into the data descriptor in the memory 23 of the receiver equipment 2.

COPYRIGHT: (C)1999,JPO

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-177588

(43) 公開日 平成11年(1999) 7月2日

(51) Int.Cl.<sup>8</sup>

識別記号

F I

H 0 4 L 12/40

G 0 6 F 13/00

13/38

H 0 4 L 12/56

29/06

3 5 7

3 5 0

H 0 4 L 11/00

G 0 6 F 13/00

13/38

H 0 4 L 11/20

13/00

3 2 0

3 5 7 A

3 5 0

1 0 2 A

3 0 5 B

審査請求 未請求 請求項の数12 O L (全 10 頁)

(21) 出願番号

特願平9-339576

(22) 出願日

平成9年(1997)12月10日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 川村 晴美

東京都品川区北品川6丁目7番35号 ソニ

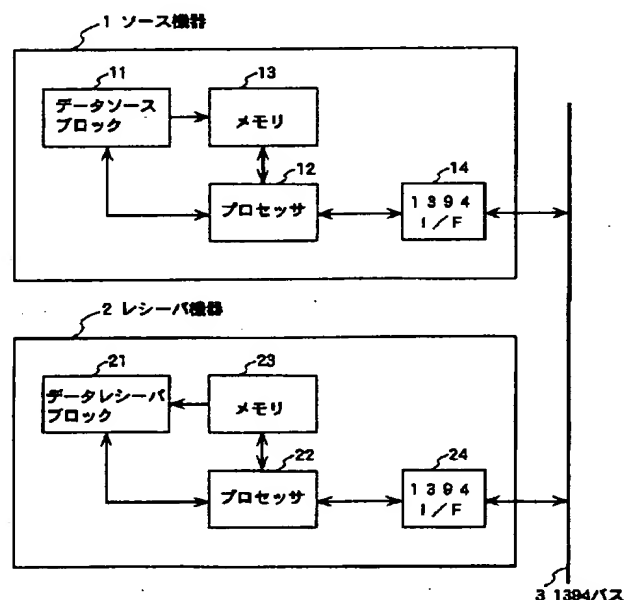
ー株式会社内

(54) 【発明の名称】 電子機器及びデータ通信方法

(57) 【要約】

【課題】 静止画像データ等のようなデータ量の少ないデータをアシンクロナス packets で伝送する。

【解決手段】 ソース機器1は、データソースブロック11から選択可能なデータをメモリ13内のデータディスクリプタに書き込む。そして、そのデータディスクリプタから読み出したデータをアシンクロナス packets でレシーバ機器2へ送信する。ソース機器2は、データソースブロック11から選択可能なデータをレシーバ機器2のメモリ23内のデータディスクリプタに書き込むこともできる。



## 【特許請求の範囲】

【請求項1】 アイソクロナス通信及びアシンクロナス通信が可能なバスで接続された複数の電子機器の間で通信を行うシステムに用いる電子機器であって、アシンクロナス通信の対象であるデータが書き込まれるデータディスクリプタ備えることを特徴とする電子機器。

【請求項2】 前記データディスクリプタにはデータソースから選択可能な複数のコンテンツのデータが書き込まれる請求項1に記載の電子機器。

【請求項3】 前記データディスクリプタにはデータソースから選択されたコンテンツのデータが書き込まれる請求項1に記載の電子機器。

【請求項4】 前記電子機器はデータの送信側の機器として構成されている請求項1に記載の電子機器。

【請求項5】 前記電子機器はデータの受信側の機器として構成されている請求項1に記載の電子機器。

【請求項6】 アイソクロナス通信及びアシンクロナス通信が可能なバスで接続された複数の電子機器の間で通信を行うシステムにおいて、送信側の電子機器は、該電子機器内のデータソースから選択可能なデータをデータディスクリプタに書き込み、該データディスクリプタから読み出したデータをアシンクロナス通信で送信することを特徴とするデータ通信方法。

【請求項7】 前記データディスクリプタに前記データソースから選択可能な複数のコンテンツのデータを書き込んだ後、受信側の電子機器が選択したコンテンツのデータを前記データディスクリプタから読み出して送信する請求項6に記載のデータ通信方法。

【請求項8】 前記送信側の電子機器は、前記受信側の電子機器が選択したコンテンツのデータを前記データディスクリプタに書き込んだ後に読み出して送信する請求項6に記載のデータ通信方法。

【請求項9】 前記データの送信中に前記データディスクリプタに書き込まれたデータが変化した場合には、それを受信側の電子機器に通知する請求項6に記載のデータ通信方法。

【請求項10】 アイソクロナス通信及びアシンクロナス通信が可能なバスで接続された複数の電子機器の間で通信を行うシステムにおいて、送信側の電子機器は、受信側の電子機器内のデータディスクリプタに、送信側の電子機器内のデータソースから選択可能なコンテンツのデータを、アシンクロナス通信で送信することを特徴とするデータ通信方法。

【請求項11】 前記送信側の電子機器は、該電子機器内のデータソースから選択可能な複数のデータを送信する請求項10に記載のデータ通信方法。

【請求項12】 前記送信側の電子機器は、該電子機器内のデータソースから選択された一つのデータを送信す

る請求項10に記載のデータ通信方法。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、IEEE1394シリアルバスのようなアイソクロナスパケットとアシンクロナスパケットとを伝送できるバスで複数の電子機器を接続し、それらの電子機器の間で通信を行うシステムに関し、詳細には静止画像データ、テキストデータ、アイコンビットマップデータ等のデータ量の少ないデータを効率良く伝送する技術に関する。

## 【0002】

【従来の技術】デジタルビデオカセットレコーダ、テレビジョン受像機、パーソナルコンピュータ等の電子機器（以下機器という）をIEEE1394シリアルバス（以下1394バスという）で接続し、これらの機器の間でビデオデータやオーディオデータ等の情報信号、及び機器の動作制御コマンドや接続制御コマンド等の制御信号を通信するシステムが考えられている。

【0003】複数の機器を1394バスで接続したシステムにおいては、パケットの伝送は所定の通信サイクル（例、125 $\mu$ sec）毎に時分割多重により行われる。このパケット伝送はサイクルマスターと呼ばれる機器が通信サイクルの開始であることを示すサイクルスタートパケットを1394バス上に送出することにより開始される。一通信サイクル中における通信の形態は、ビデオデータやオーディオデータ等の情報信号をリアルタイムで伝送するアイソクロナス（以下Isoという）通信と、機器の動作制御コマンドや接続制御コマンド等の制御信号を必要に応じて不定期に伝送するアシンクロナス（以下Asyncという）通信の二種類がある。そして、IsoパケットはAsyncパケットより先に伝送される。Isoパケットのそれぞれにチャンネル番号を付けることにより、複数のIsoデータを区別することができる。送信すべき全てのIsoパケットの伝送が終了した後、次のサイクルスタートパケットまでの期間がAsyncパケットの伝送に使用される。

【0004】前述したような1394バスで複数の機器を接続したシステムにおいて、オーディオ・ビデオ/コントロール ディスクリプタモデル (AV/C descriptor model) により情報信号を扱うことが提案されている（特願平9-96874号、特願平9-177917号）。すなわち、例えばデジタルチューナーのサービスリスト、ディスクプレーヤーのトラックリストのように、データソースから選択可能なコンテンツを図13に示す構造を有するオブジェクトを用いてオブジェクトリストという形式で機器内のメモリに保持しておき、そのオブジェクトリストから希望するオブジェクトを指定すると、そのオブジェクトのデータストリームが機器からIsoパケットを用いて出力される。

## 【0005】

【発明が解決しようとする課題】1394バスで接続された機器の間で静止画像データ、テキストデータ、アイコンビットマップデータ等の静的なデータを伝送することが考えられている。これらのデータのリストを前述したオブジェクトリストとして機器内に保持しておき、選択されたデータをIsoパケットを用いて伝送することは勿論可能であるが、これらのデータはオーディオデータやビデオデータ等の同期信号を必要とするデータとは異なり、データ量も少なく、Iso通信で伝送するメリットがない。しかし、これらのデータをAV（オーディオ・ビデオ）機器の間でAsync通信を用いて伝送する仕組みは考えられていなかった。

【0006】本発明はこのような問題点に鑑みてなされたものであって、前述した静的なデータのようなデータ量の少ないデータをAsync通信で伝送する手段を提供するものである。

【0007】

【課題を解決するための手段】本発明に係る機器は、Iso通信及びAsync通信が可能なバスで接続された複数の機器の間で通信を行うシステムに用いる機器であって、Async通信の対象であるデータが書き込まれるデータディスクリプタを備えることを特徴とするものである。

【0008】本発明に係るデータ通信方法は、Iso通信及びAsync通信が可能なバスで接続された複数の機器の間で通信を行うシステムにおいて、送信側の機器は、その機器内部のデータソースから選択可能なデータをデータディスクリプタに書き込み、そのデータディスクリプタから読み出したデータをAsync通信で送信することを特徴とするものである。

【0009】また、本発明に係るデータ通信方法は、Iso通信及びAsync通信が可能なバスで接続された複数の機器の間で通信を行うシステムにおいて、送信側の機器は、受信側の機器内のデータディスクリプタに、送信側の機器内のデータソースから選択可能なコンテンツのデータを、Async通信で送信することを特徴とするものである。

【0010】本発明によれば、送信側の機器は、その機器内部のデータソースから選択可能なデータをデータディスクリプタに書き込み、そのデータディスクリプタから読み出したデータをAsync通信で送信する。

【0011】また、本発明によれば、送信側の機器は、受信側の機器内のデータディスクリプタに、送信側の機器内のデータソースから選択可能なコンテンツのデータを、Async通信で送信する。

【0012】

【発明の実施の形態】以下、本発明の実施の形態について図面を参照しながら詳細に説明する。

【0013】図1は本発明を適用した通信システムの構成を示すブロック図である。この通信システムは、ソー

ス機器1とレシーバ機器2とが1394バス3で接続されており、ソース機器1からレシーバ機器2に対して静止画像データ、テキストデータ、アイコンビットマップデータ等の静的なデータを送信することができる。

【0014】ソース機器1は、内部にデータソースブロック11とプロセッサ12とメモリ13と1394インターフェース14とを備えている。データソースブロック11は前述した静的なデータ（以下の説明では静止画像データを例示する）を発生する。プロセッサ12はこのソース機器1全体の制御等を行う。メモリ13はRAM等で構成されており、データソースブロック11が発生した静止画像データ及びその属性データ等を書き込むためのエリアが設けられている。1394インターフェース14はプロセッサ12がメモリ13のディスクリプタから読み出した静止画像データやプロセッサ12が作成した各種コマンド/レスポンス等をAsyncパケットに格納して1394バス3に送出する。また、レシーバ機器2が1394バス3に送出したAsyncパケットから各種コマンド/レスポンス等を取り出してプロセッサ12に送る。

【0015】同様に、レシーバ機器2は、内部にデータレシーバブロック21とプロセッサ22とメモリ23と1394インターフェース24とを備えている。1394インターフェース24はソース機器1が1394バス3に送出したAsyncパケットを受信して、そこから静止画像データや各種コマンド/レスポンス等を分離しプロセッサ22に送る。プロセッサ22は1394インターフェース24から受け取った静止画像データをメモリ23に書き込む。また、各種コマンド/レスポンス等を解釈してそれに対応する処理を実行する。さらに、このレシーバ機器2全体の制御等を行う。メモリ23はRAM等で構成されている。データレシーバブロック21はプロセッサ22の制御に従って、メモリ23内の静止画像データを取り込む。

【0016】なお、実際にはユーザーが各種指令等を入力するための入力手段（キーボード、リモートコンマンド等）やディスプレイ手段等が設けられているが、ここでは省略した。

【0017】図2は図1のメモリ13内に設けられるコンテンツリスト（Contents List）及びデータディスクリプタの構成例を示す。コンテンツリスト41の基本構成は図13に示した。ここではアドレス“0002”のリストタイプを“静止画像”とした。また、エントリー数（number\_of\_entries）は“n”とした。各エントリー（entry）は1枚の静止画像に対応して作成される。ここでは、n枚の静止画像がエントリーされているので、データソースブロック11はn枚の静止画像を発生する。各エントリーにはそのエントリーの子リスト（Child List）のリストIDとそのエントリーの静止画像の属性等が書き

10

20

30

40

50

込まれている。ここでは、エンタリー [0] の子リストであるデータディスクリプタ (Data\_Descriptor) 42とエンタリー [n-1] の子リストであるデータディスクリプタ43を図示した。

【0018】図3に今回新たに提案したデータディスクリプタの構造の一例を示す。AV/C オブジェクトリストモデルでは、オブジェクトエンタリー (object\_entry) の下の階層としてそのオブジェクトに関連した情報のリストであるオブジェクトリスト (object\_list) を持つことができる。したがって、このデータディスクリプタもリストID (list\_id) を持つことはオブジェクトリストと同じである。ただし、オブジェクトリストには選択可能なコンテンツのリストが格納されていたものであるが、データディスクリプタにはコンテンツ (ここでは静止画像データ) そのものが格納されている。また、ディスクリプタ長 (descriptor\_length) のフィールドを3バイトとし16Mバイトまでのデータを扱えるようにした。さらに、従来のオブジェクトリストと互換性を持ち、かつ新たなディスクリプタとして活用できるようにするために、従来のオブジェクトリストにおけるディスクリプタ長フィールドに相当するフィールド (アドレス“00 00 00”と“00 00 01”) に格納する値を“FFFF”とし、リストタイプ (アドレス“00 00 02”に格納) をデータディスクリプタとし、それ以降のフィールドを新たに定義している。

【0019】アドレス“00 00 03”のステータス (status) は、データディスクリプタに格納されているデータのステータス (valid/invalid, stable/busy 等) を表す。アドレス“00 00 05”~“00 00 07”の3バイトにはデータディスクリプタのディスクリプタ長を格納する。そして、アドレス“00 00 08”以降にはデータ (ここでは静止画像データ) を格納する。

【0020】1394バスに接続された機器内に設けられたディスクリプタの内容を1394バスに接続された他の機器が読み出すためには、ディスクリプタ読み出し (READ\_DESCRIPTOR) コマンドを使用する。したがって、図1におけるレシーバ機器2がソース機器1内のメモリ13内に作成されたコンテンツリスト41及びデータディスクリプタ42, 43の内容を読み出すためには、レシーバ機器2はディスクリプタ読み出しコマンドを1394バス3を介してソース機器1へ送ることが必要である。ソース機器1はこのディスクリプタ読み出しコマンドを受け取ると、メモリ13からディスクリプタの内容を読み出し、それをレスポンスとしてレシーバ機器2へ返送する。同様に、他の機器がディスクリプタに書き込みを行うためには、ディスクリプタ書き込み (WRITE\_DESCRIPTOR) コマンドを使用する。

【0021】図4はディスクリプタ読み出しコマンドの構成の一例である。このディスクリプタ読み出しコマンドは、図3に示したデータディスクリプタの読み出しを可能にするために新たに提案したものである。すなわち、オペランド [4] ~ [5] に格納されている2バイトのデータ長を“FFFF”とし、かつオペランド [6] ~ [7] に格納されている2バイトの読み出しアドレスを“FFFF”としている。そして、オペランド [8] ~ [10] の3バイトで読み出し対象となるデータディスクリプタの長さを表し、オペランド [11] ~ [13] の3バイトでデータディスクリプタの読み出しアドレスを表している。

【0022】同様に、図5はディスクリプタ書き込みコマンドの構成の一例である。ディスクリプタ書き込みコマンドでは、オペランド [14] 以降に書き込みを行うデータが格納される。

【0023】これらのディスクリプタ読み出しコマンド又はディスクリプタ書き込みコマンドを使用することにより、図3のBに示した16Mバイトの領域をアクセスすることができるようになる (従来のディスクリプタ読み出しコマンド又はディスクリプタ書き込みコマンドでは、図3のAに示した64Kバイトのアクセスのみ可能)。

【0024】図1のレシーバ機器2が図1のソース機器1におけるデータディスクリプタの読み出しを行っている間に、ソース機器1内でデータディスクリプタ内のデータが書き換わってしまった場合には、読み出されたデータは保証されない。そこで、ソース機器1のプロセッサ12はデータディスクリプタのアドレス“00 00 03”のステータスフィールドにデータディスクリプタ内のデータの状態を表す情報 (stable/busy 等) を書き込んでおく。そして、レシーバ機器2はソース機器1に対して、データディスクリプタに変化があった時に知らせてもらうためのコマンドである通知 (NOTIFY) コマンドを送っておく。この時に使用する通知コマンドのオペレーションコードとオペランドを図6 (a) に示す。また、ソース機器1からレシーバ機器2に返送されるレスポンスのオペレーションコードとオペランドを図6 (b) に示す。次に図7を参照しながら、図1に示した通信システムにおいてレシーバ機器2がソース機器1内のデータディスクリプタから静止画像データを読み出す場合の動作を説明する。なお、図7以降のフローチャートにおいて、送信されたコマンドに対して当然レスポンスが返送されるが、特に必要な場合を除いてレスポンスの内容については説明しない。

【0025】まず、レシーバ機器2はソース機器1に対してコンテンツリスト41を開くためのコマンド (OPEN\_DESCRIPTOR Contents List) を送信する (ステップS1)。これにより、レシーバ機器2はコンテンツリスト41にアクセスすること

が可能となる。

【0026】次に、レシーバ機器2はソース機器1に対してコンテンツリスト41を読み出すためのコマンド (READ DESCRIPTOR Contents List) を送信する (ステップS2)。ソース機器1はこのコマンドを受け取ると、メモリ13からコンテンツリスト41を読み出し、レスポンスに格納してレシーバ機器2に返送する。

【0027】次に、レシーバ機器2はレスポンスに格納されているコンテンツリストのエントリから、データディスクリプタのリストIDを読み出す (ステップS3)。そして、受信を希望する静止画像データが書き込まれているデータディスクリプタのリストIDをメモリ23に保存しておく。

【0028】次に、レシーバ機器2はソース機器1に対してコンテンツリスト41を閉じるためのコマンド (CLOSE DESCRIPTOR Contents List) を送信する (ステップS4)。

【0029】次に、レシーバ機器2はソース機器1に対して、受信を希望する静止画像が書き込まれているデータディスクリプタ (ここではデータディスクリプタ42とする) を開くためのコマンド (OPEN DESCRIPTOR Data Descriptor) を送信する (ステップS5)。これにより、レシーバ機器2はデータディスクリプタ42にアクセスすることが可能となる。

【0030】さらに、レシーバ機器2はソース機器1に対して、データディスクリプタ42の内容が変化した時にそれを通知してもらうコマンド (NOTIFY DataDescriptor Status) を送信する (ステップS6)。

【0031】次に、レシーバ機器2はソース機器1からデータディスクリプタ42の内容に変化があったことの通知のレスポンスがあるかどうか判断し (ステップS7)、それがあった場合にはデータディスクリプタ42内のデータが保証されないので、エラー処理 (ステップS11) を行った後に処理を終了する。

【0032】ステップS7で通知がなかった場合には、レシーバ機器2はソース機器1に対して、データディスクリプタ42を読み出すためのコマンド (READ DESCRIPTOR Data Descriptor) を送信する (ステップS8)。ソース機器1はこのコマンドを受け取ると、データディスクリプタ42から静止画像データを読み出しレスポンスに格納してレシーバ機器2に返送する。

【0033】次に、レシーバ機器2はデータディスクリプタ42内の全ての静止画像データの読み出しが完了したかどうかを判断し、完了していなければステップS7からの処理を繰り返す。なお、ステップS9の処理の詳細については後述する。

【0034】ステップS9で全ての静止画像データの読み出しが完了したと判断した場合には、ソース機器1に対して、データディスクリプタ42を閉じるためのコマンド (CLOSE DESCRIPTOR Data Descriptor) を送信 (ステップS10) した後、処理を終了する。

【0035】図8は図7におけるデータディスクリプタの内容を読み出す処理の詳細を示すフローチャートである。

10 【0036】まず、データディスクリプタのアドレス “00 00 05” ~ “00 00 07” に格納されている3バイトのディスクリプタ長を見て全データサイズを調べる (ステップS21)。ここでは、全データサイズがxバイトであったものとする。

【0037】次に、レシーバ機器2自身が3バイト対応であるかどうか、すなわち図3に示したデータディスクリプタに対応しているかどうかを判断する (ステップS22)。対応しているかどうかは、例えば予めレシーバ機器2内のROM (図示せず) に格納されている情報をプロセッサ22が読み出すことで判断する。

【0038】ステップS22で3バイト対応であると判断したら、次に読み出しアドレスyを “00 00 08” に設定し (ステップS24)、さらにyとxの大小関係判断 (ステップS25) する。そして、yがx以下であれば、アドレスyからzバイトを読み出す。すなわち、レシーバ機器2は図4に示したディスクリプタ読み出しコマンドのオペランド [8] ~ [10] のデータ長をzバイトとし、オペランド [11] ~ [13] のアドレスをyにしてソース機器1に送信する。

30 【0039】ソース機器1はこのコマンドを受け取ると、z1バイト ( $z1 \leq z$ ) のデータをレスポンスに格納してレシーバ機器2に送る。レシーバ機器2はレスポンスのデータ長フィールドの長さがz1バイトであったので (ステップS27)、次に  $z = z1$  に設定し (ステップS28)、さらにyを  $y + z$  にインクリメントした (ステップS29) 後にステップS25に戻る。つまり、レシーバ機器2が指定したサイズのデータをソース機器1が送信できない場合には、ソース機器1は送信できるサイズのデータを送り、レシーバ機器2は次回からはソース機器1が送信できるサイズのデータを要求する。

【0040】ステップS25 ~ S29の処理を繰り返し、 $y > x$  になったら (ステップS25でYES)、全てのデータが読み出されたと判断して処理を終了する。

【0041】なお、ステップS22で3バイト対応でない判断した場合には、全データサイズxが64Kバイトより大きいかどうか判断する (ステップS23)。そして、64Kバイト以下であれば、読み出すことが可能なので、ステップS24に進む。全データサイズが64Kバイトより大きい場合には、読み出すことができない

ので処理を終了する。

【0042】以上の説明はメモリ13内のコンテンツリスト41における全てのエントリーに対してデータディスクリプタが作成されている場合の例であった。次に、レシーバ機器2が選択したエントリーのデータディスクリプタのみがコンテンツリスト41の子リストとして作成される場合の処理について、図9を参照しながら説明する。なお、図9において図7と同一の処理には図7で使したステップ番号と同一のステップ番号を付した。

【0043】図9の処理において、最初の2個のステップは図7と同一である。そして、3番目のステップS3'では、レシーバ機器2はソース機器1に対して、コンテンツリストを閉じるためのコマンド(CLOSE DESCRIPTOR Contents List)を送信する。

【0044】次に、レシーバ機器2はソース機器1に対して、オブジェクト(コンテンツリスト41におけるエントリー)を選択するためのコマンドを送信する(ステップS4')。ソース機器1はこのコマンドを受け取ると、選択されたエントリーの子リストとしてデータディスクリプタを作成する。これ以降のレシーバ機器2の処理は図7で説明した処理と同様である。

【0045】これまでの説明はソース機器1内のメモリ13にデータディスクリプタを作成するものであった。次に、レシーバ機器2内のメモリ23にデータディスクリプタを作成する場合について説明する。

【0046】図10はメモリ23内にコンテンツリストの全てのエントリーのデータディスクリプタを作成する場合のソース機器1の処理を示すフローチャートである。

【0047】まず、ソース機器1は、レシーバ機器2に対してそのメモリ23内のコンテンツリストを開くためのコマンド(OPEN DESCRIPTOR Contents List)を送信する(ステップS31)。

【0048】次いで、ソース機器1は、レシーバ機器2に対して、メモリ23内のコンテンツリストを読み出すためのコマンド(READ DESCRIPTOR Contents List)を送信する(ステップS32)。

【0049】次に、ソース機器1は、コンテンツリスト読み出しコマンドに対するレスポンスに格納されているコンテンツリストのエントリーから、データディスクリプタのリストIDを読み出した(ステップS33)後、そのコンテンツリストを閉じるためのコマンド(CLOSE DESCRIPTOR Contents List)を送信する(ステップS34)。

【0050】次に、ソース機器1は、レシーバ機器2に対してデータディスクリプタを開くためのコマンド(OPEN DESCRIPTOR Data Descr

iptor)を送信する(ステップS35)。このコマンドには、静止画像データを書き込む対象であるデータディスクリプタのリストIDを格納しておく。

【0051】次に、ソース機器1は、レシーバ機器2に対してデータディスクリプタに、自分のデータソースブロック11の静止画像データを書き込むためのコマンド(WRITE DESCRIPTOR Data Descriptor)を送信する(ステップS36)。

【0052】そして、ソース機器1はレシーバ機器2内のデータディスクリプタに全ての静止画像データを書き込みが完了したら(ステップS37でYES)、レシーバ機器2に対してそのデータディスクリプタを閉じるためのコマンド(CLOSE DESCRIPTOR Data Descriptor)を送信し(ステップS38)、処理を終了する。

【0053】なお、レシーバ機器2において、メモリ23への書き込み動作(データディスクリプタへの書き込み)が追いつかない場合には、データディスクリプタの書き込みコマンドに対して拒絶(REJECT)のレスポンスを返す。

【0054】図11はデータソースブロック11から選択されている単一のコンテンツをレシーバ機器内の単一のデータディスクリプタに書き込む場合のレシーバ機器2の処理を示すフローチャートである。

【0055】まず、レシーバ機器2は、メモリ23内のデータディスクリプタを開くためのコマンド(OPEN DESCRIPTOR Data Descriptor)をソース機器1から受信する(ステップS41)。これにより、ソース機器1はメモリ23内のデータディスクリプタにアクセスすることが可能になる。

【0056】次に、レシーバ機器2はメモリ23内のデータディスクリプタにデータの書き込みを行うためのコマンド(WRITE DESCRIPTOR Data Descriptor)をソース機器1から受信する(ステップS42)。そして、全てのデータの書き込みを完了するまで、ステップS42の処理を繰り返す(ステップS43)。全てのデータの書き込みが完了したら、レシーバ機器2はデータディスクリプタを閉じるためのコマンド(CLOSE DESCRIPTOR Data Descriptor)をソース機器1から受信した(ステップS44)後、データディスクリプタのデータ処理を行う(ステップS45)。データディスクリプタの処理としては、例えば図12に示すように、データディスクリプタに書き込まれたデータを読み出し、そのデータ中の時刻情報を用いてリアルタイムのデータストリームを作成することが考えられる。なお、データディスクリプタに書き込まれたデータをどのように処理するかはレシーバ機器2が管理する。

【0057】図10～図12を参照しながら説明したシステムでは、レシーバ機器2の処理速度が間に合えば、



20kbp/s程度の低レートのオーディオデータ等の動的なデータの伝送に適用することも可能である。

#### 【0058】

【発明の効果】以上詳細に説明したように、本発明によれば静的なデータや低レートのオーディオデータのようなデータ量の少ないデータをAsyncパケットを用いて伝送することが可能である。

【0059】また、このようなデータをAsyncパケットで伝送するために、従来のディスクリプタとコマンド/レスポンスのフォーマットを拡張するだけでよいので、新たなプロトコルを開発・実装する必要がない。

#### 【図面の簡単な説明】

【図1】本発明を適用した通信システムの構成を示すブロック図である。

【図2】図1のメモリ内に設けられるコンテンツリスト及びデータディスクリプタの構成例を示す図である。

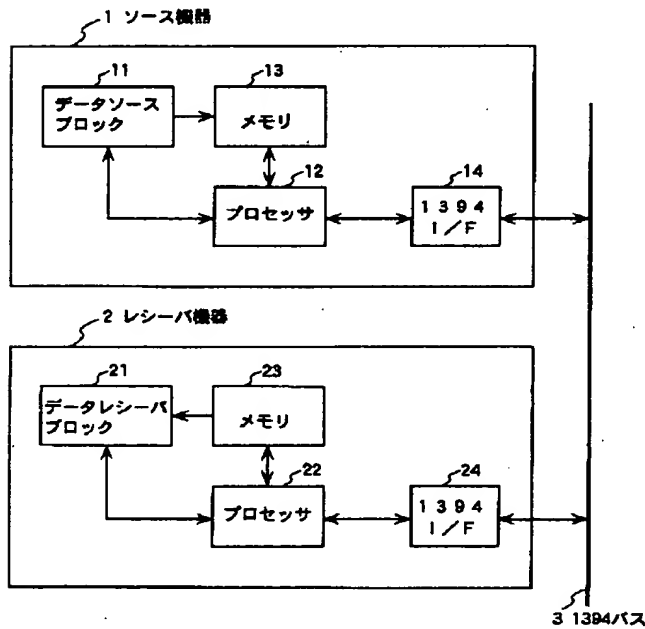
【図3】今回新たに提案したデータディスクリプタの構造の一例を示す図である。

【図4】ディスクリプタ読み出しコマンドの構成の一例を示す図である。

【図5】ディスクリプタ書き込みコマンドの構成の一例を示す図である。

【図6】データディスクリプタのステータスの変化を通

【図1】



知するためのコマンド/レスポンスの構成の一例を示す図である。

【図7】図1に示した通信システムにおいてレシーバ機器がソース機器内のディスクリプタ内の静止画像データを読み出す場合の動作を示す図である。

【図8】図7におけるデータディスクリプタの内容を読み出す処理の詳細を示すフローチャートである。

【図9】レシーバ機器が選択したエンタリーのデータディスクリプタのみが作成される場合の処理を示すフローチャートである。

【図10】レシーバ機器内に全てのエンタリーのデータディスクリプタを作成する場合のソース機器の処理を示すフローチャートである。

【図11】レシーバ機器内に予め選択されたエンタリーのデータディスクリプタを作成する場合の処理を示すフローチャートである。

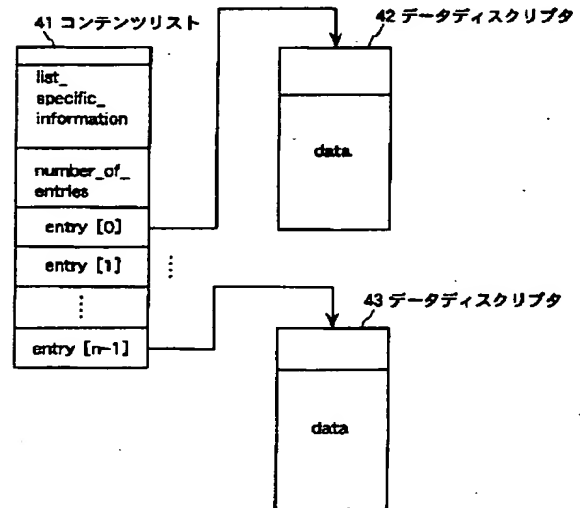
【図12】レシーバ機器内のデータディスクリプタに書き込まれたデータの処理の例を示す図である。

【図13】オブジェクトリストの構造を示す図である。

#### 20 【符号の説明】

1…ソース機器、2…レシーバ機器、3…1394バス、41…コンテンツリスト、42…データディスクリプタ。

【図2】



【図6】

(a)	opcode	DATA DESCRIPTOR STATUS
	operand [0]	FF

(b)	opcode	DATA DESCRIPTOR STATUS
	operand [0]	status (stable/busy)

【図 3】

address	contents
00 00 00	descriptor_length (FF)
00 00 01	(FF)
00 00 02	list_type (Data Descriptor)
00 00 03	status
00 00 04	reserved
00 00 05	descriptor_length
00 00 06	data
00 00 07	
00 00 08	
:	
:	
:	
:	
:	
:	
:	
00 FF FF	
01 00 00	
:	
:	
FF FF FF	

A : 64 Kバイト

B : 16 Mバイト

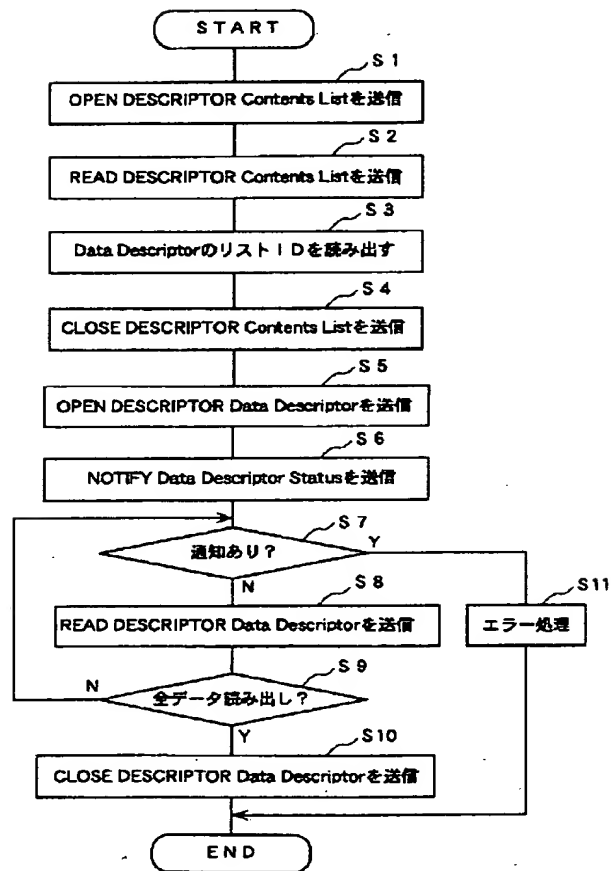
【図 4】

opcode	READ DESCRIPTOR
operand [0]	descriptor_identifier
:	[1]
:	[2]
:	[3] read_result_status
:	[4] data_length (FF)
:	[5] (FF)
:	[6] address (FF)
:	[7] (FF)
:	[8] data_length for Data Descriptor
:	[9]
:	[10]
:	[11] address for Data Descriptor
:	[12]
:	[13]

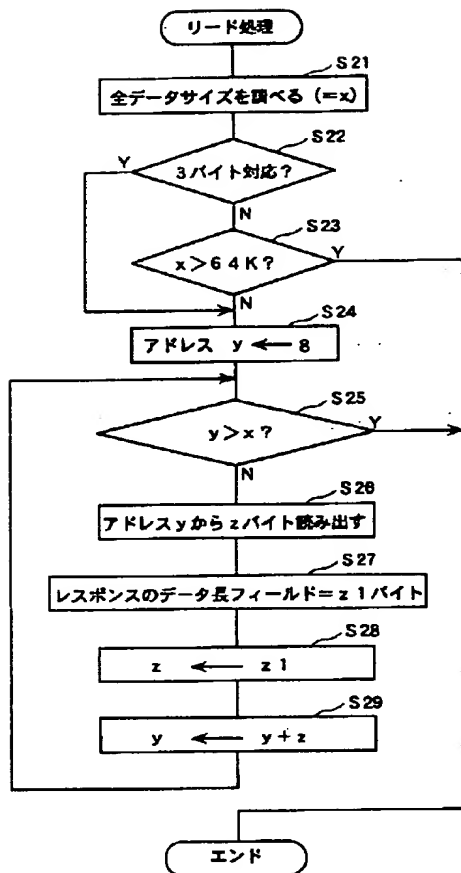
【図 5】

opcode	WRITE DESCRIPTOR
operand [0]	descriptor_identifier
:	[1]
:	[2]
:	[3] subfunction
:	[4] data_length (FF)
:	[5] (FF)
:	[6] address (FF)
:	[7] (FF)
:	[8] data_length for Data Descriptor
:	[9]
:	[10]
:	[11] address for Data Descriptor
:	[12]
:	[13]
:	[14] data
:	
:	
:	
:	
:	

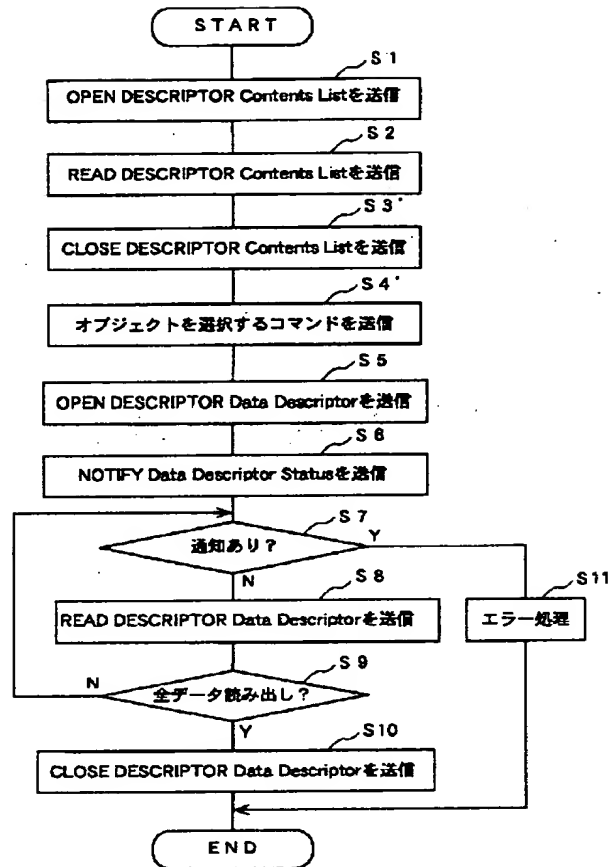
【図 7】



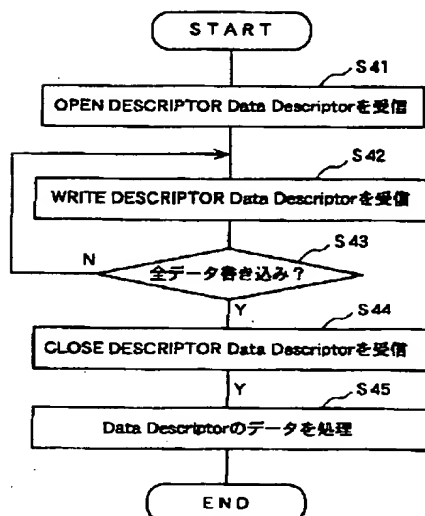
【図8】



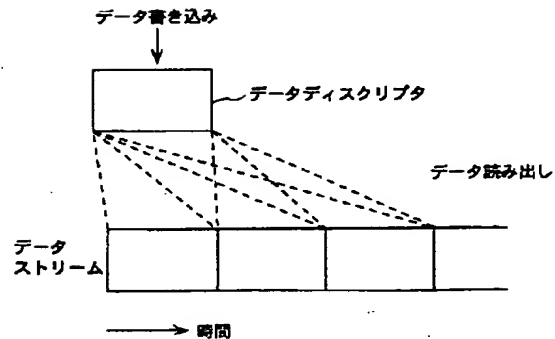
【図9】



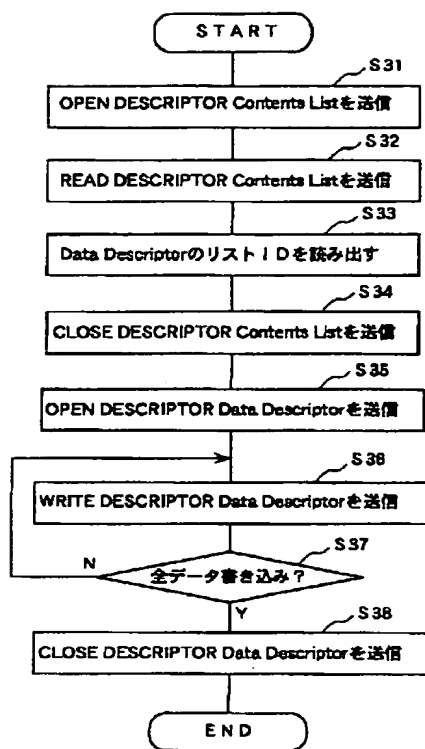
【図11】



【図12】



【図 10】



【図 13】

address	contents
00 00	descriptor_length
00 01	
00 02	list_type
00 03	attributes
00 04	list_specific_information
:	
:	
:	number_of_entries(n)
:	
:	object_entry [0]
:	
:	:
:	object_entry [n-1]
:	
:	